

REMARKS

Claims 1-21 are pending in the application. In response to the Office Action, applicants have amended independent claims 1 and 11. Claims 1-21 remain in this application and are now pending for reconsideration.

Claim 1 has been amended to clarify that the “mutual exclusion” recited in that claim is achieved between multiple threads performing work on the shared resource and a single thread updating or changing the state of the shared resource without requiring serialization of all threads such that an update or change of the state of the shared resource may be made by the single thread only when none of the multiple threads are processing work on the shared resource.

Similarly, claim 11 has been amended to clarify that the “mutual exclusion” recited in that claim is achieved between multiple threads performing work on the work queues and a single thread changing the state of the work queues without requiring serialization of all threads such that an update or change of the state of the work queues may be made by the single thread only when none of the multiple threads are processing work on the work queues. These amendments to claims 1 and 11 have been made to ensure a clear meaning of the different uses of “mutual exclusion” in those claims, and were not made for patentability reasons.

Claims 1-3 and 9-10 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg et al. (U.S. Patent No. 5,319,782) in view of Valtin et al. (U.S. Patent No. 6,243,107). Claims 4-5 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg et al. (U.S. Patent No. 5,319,782) in view of Valtin et al. (U.S. Patent No. 6,243,107) as applied to claim 1, and further in view of Vishkin (U.S. Patent No. 6,461,527). Claims 6-7 and 20-21 were

rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg et al. (U.S. Patent No. 5,319,782) in view of Valtin et al. (U.S. Patent No. 6,243,107) in view of Vishkin (U.S. Patent No. 6,461,527) and further in view of Haber et al. (U.S. Patent No. 4,435,766). Claims 8, 11-12 and 17-19 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg et al. (U.S. Patent No. 5,319,782) in view of Valtin et al. (U.S. Patent No. 6,243,107) and further in view of Tillier (U.S. Patent No. 6,421,742). Claims 13-14 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg et al. (U.S. Patent No. 5,319,782) in view of Valtin et al. (U.S. Patent No. 6,243,107) in view of Tillier (U.S. Patent No. 6,421,742) and further in view of Vishkin (U.S. Patent No. 6,461,527). Claims 15-16 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg et al. (U.S. Patent No. 5,319,782) in view of Valtin et al. (U.S. Patent No. 6,243,107) in view of Tillier (U.S. Patent No. 6,421,742) in view of Vishkin (U.S. Patent No. 6,461,527) and further in view of Haber et al. (U.S. Patent No. 4,435,766).

The Applicants respectfully traverse the assertions that the various combinations of the five references relied upon in the various rejections would have been obvious. For example, the Applicants respectfully submit that it would not have been obvious to combine the Goldberg et al. and the Valtin et al. patents in the manner suggested by the Examiner, except in hindsight in view of the present invention. Goldberg et al. discloses a thread that becomes available on a work queue of threads for binding to a task only if the thread owns exactly one resource, and after execution, ownership of the resource is relinquished and passed to the next thread queued on that resource. Valtin et al. discloses a system with at least one application thread or master thread, and in addition a slave thread assigned to each CPU in the system. No shared resources

are mentioned in Valtin et al. It is difficult to understand how one of ordinary skill in the art would have made the jump from a system including a resource as disclosed in Goldberg et al. where only one thread owns exactly one resource to a system as disclosed in Valtin et al. where each CPU in a system has one dedicated slave thread.

In addition, even if one of ordinary skill in the art were motivated to combine these two references the Applicants respectfully submit that such a combination would not have rendered the claims of this application obvious. In the present invention as claimed, for example, in claim 1, an operating system is configured to allow multiple processors to perform work on a shared resource concurrently while supporting state changes or updates of the shared resource. In addition claim 1 recites a synchronization algorithm for synchronizing multiple threads of operation with a single thread to achieve mutual exclusion. Goldberg et al. discloses exclusion of all threads and no concurrent work being performed on a resource. Valtin et al. does not disclose mutual exclusion of all work threads and a single thread. Valtin does not disclose any concurrent work performed on a resource. In fact Valtin et al. discloses that when one thread is working on a CPU, no other thread can work (see column 3, lines 13-15).

Although one of ordinary skill in the art would never have thought to combine the systems of Goldberg et al. and Valtin et al., even if they had combined the systems they might have come up with a system with multiple CPUs and a resource, where only one thread worked on the resource at one time, or a system with one master thread executing an application to generate data for respective slave threads in respective CPUs. The Applicants respectfully submit that each of the Goldberg et al. and Valtin et al. references actually teach away from the

present invention as claimed in claim 1. Similarly, these references also teach away from combining them to render claim 11 obvious. None of the references relied upon by the Examiner either alone or in combination disclose or suggest at least the features of the present invention of concurrently performing of work on a shared resource or a work queue, achieving mutual exclusion and/or allowing an update thread to access and update a shared resource or a work queue as specifically claimed in the claims of the present application.

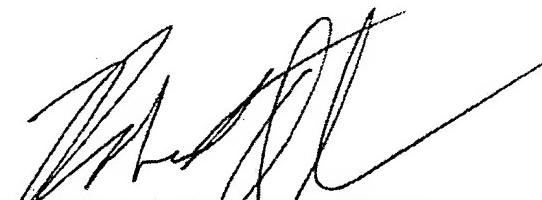
The Applicants also respectfully submit that one of ordinary skill in the art would not have added the Vishkin and Haber references to render claims 20 and 21 obvious. The Applicants respectfully traverse the assertion that it would have been obvious to add the lock in Haber to the systems in the other patents. In addition, even it were obvious to add some sort of lock feature to a combination of Goldberg et al., Valtin et al and Vishkin, one never would have come up with a system that included at least the claimed feature of allowing an update thread to access and update when no worker threads are running, except in hindsight in view of the present invention.

The Applicants respectfully traverse the rejections in the outstanding Office Action for at least the reasons set forth above.

In view of the foregoing, the application is considered to be in condition for allowance.

Early notification of the same is earnestly solicited. If there are any questions regarding the present application, the Examiner is invited to contact the undersigned attorney at the telephone number listed below.

Respectfully submitted,



Robert D. Anderson
Reg. No. 33,826
(815) 633 - 2563

August 25, 2003

Date

Intel Americas, Inc.
2109 Shaw Woods Drive
Rockford, Illinois 61107

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313 on:

8/25/03

Date of Deposit

Annie Pearson

Name of Person Mailing Correspondence

Annie Pearson
Signature

8/25/03
Date